# Container (data structure)

From Wikipedia, the free encyclopedia

In computer science, a **container** is a class, a data structure, or an abstract data type (ADT) whose instances are collections of other objects. In other words; They are used to store objects in an organized way following specific access rules.

Generally, container classes are expected to implement methods to do the following:

- create a new empty container (constructor),
- report the number of objects it stores (size),
- delete all the objects in the container (clear),
- insert new objects into the container,
- remove objects from it,
- provide access to the stored objects.

There are two types of containers: value containers and reference containers.

*Value* based containers store copies of the objects. Accessing an object also returns a copy of it. Modifying an external object after it has been inserted in the container will *not* affect the content of the container.

*Reference* based containers only store pointers or references to the objects. Accessing an object returns a reference to it. Modifying an external object after it has been inserted in the container *could* result in modifying the content of the container (or more precisely, the object stored in the container).

Containers are sometimes implemented in conjunction with iterators.

# Contents

# Examples of containers

Examples of containers include:

- array
- list
- map
- queue
- set
- stack
- table
- tree
- vector

# Containers in object-oriented programming

In object-oriented programming, a **container class** is any class that is capable of storing other objects. Container

classes usually implement some kind of data structure, such as a list, map, set, array, or tree. A container class is usually able to store an arbitrary number of data items, i.e. the size of the collection is adjusted automatically.

## Graphic containers

Widget toolkits use special widgets also called *Containers* to group the other widgets together (windows, panels, ...). Apart from their graphical properties, they have the same type of behavior as **container classes**, as they keep a list of their child widgets, and allow to add, remove, or retrieve widgets amongst their children.

## Implementations

- Java: Java Collections Framework
- C++: Standard C++ library or Standard Template Library
- .NET: System.Data(MSDN) (http://msdn.microsoft.com/en-us/library/system.data.aspx)

## See also

- Standard Template Library#Containers
- dlib C++ Library
- collection (computing)

Retrieved from "http://en.wikipedia.org/wiki/Container_(data_structure)"
Categories: Object-oriented programming | Data structures